

IN THE CLAIMS:

Please cancel claims ~~1-9~~ and ~~25-30~~.

Please amend claims ~~10~~, ~~14~~, ~~16~~, ~~17~~, ~~19~~, ~~20~~ and ~~21~~ as follows

1 1. (Canceled) A system for rendering graphics primitives using a shortened display list for
2 ensuring a shortened processing time while maintaining the quality of the display information
3 contained in the display list, the system comprising:

4 a system bus for communicating data and instructions;

5 a host processor coupled to the system bus for processing a display list defining a
6 graphics primitive;

7 a system memory coupled to the system bus for storing the display list;

8 a graphics subsystem coupled to the system bus for processing display parameter values
9 contained in the display list, wherein the display list includes a field load

10 instruction for effectively shortening display list processing; and

11 a display unit coupled to the graphics processor for displaying the graphics primitives
12 comprising the display list.

1 2. (Canceled) The system of claim 1 further including a frame buffer coupled to the
2 graphics processor for storing the display list.

1 3. (Canceled) The system of claim 1, wherein the graphics subsystem includes a plurality of
2 storage means for storing the parameter values representative of the graphics primitives in the
3 display list.

1 4. (Canceled) The system of claim 3, wherein the graphics subsystem further includes a
2 address generating means for generating address offsets bits responsive to the display parameter
3 values contained in the display list.

1 5. (Canceled) The system of claim 4, wherein the graphics subsystem further includes an
2 instruction storing means for storing a plurality of instruction data bits responsive to each of the
3 plurality of display list instructions, said instruction data bits shortened to allow the display list
4 to be processed within a shortened processing cycle in the graphics subsystem.

1 6. (Canceled) The system of claim 5, wherein the graphics subsystem further includes an
2 instruction sequencing means coupled to the instruction bits storage means for sequencing bits of
3 the display list into a plurality of register files.

1 7. (Canceled) The system of claim 6, wherein the graphics subsystem further includes an
2 instruction fetch means for randomly fetching a next display list parameter value for a display
3 primitive to be processed in the graphics subsystem.

1 8. (Canceled) The system of claim 7, wherein the graphics subsystem further includes an
2 address counting means for sequentially counting the address storage locations for the display list
3 parameter values of the graphics primitives stored in the plurality of register files.

1 9. (Canceled) The system of claim 8, wherein the graphics subsystem further includes an
2 address partition storage means for storing addresses responsive to load instructions representing
3 the shortened display list, and wherein portions of the instruction storing means index the address
4 partition storage means.

1 10. (Amended) A graphics system for processing parameter values of graphics primitives in
2 a display list, wherein the display list is shortened to enable fast processing time while
3 maintaining the quality of information contained in the display list, the graphics system
4 comprising:

5 a plurality of register files for storing a plurality of parameter values representing
6 graphics primitives defined in the display list; and

7 a graphics processor coupled to the plurality of register files, wherein the graphics
8 processor processes the shortened display list while maintaining the display
9 quality of primitives displayed in a display unit, the graphics processor processing
10 load instructions representative of a shortened display list instruction including
11 the fetching of parameters associated with the shortened display list instruction
12 and storing fetched parameters randomly in the plurality of register files.

1 11. (Unchanged) The graphics processor of claim 10 including an instruction fetch logic unit
2 for fetching the next parameter values responsive to a graphics primitive to be displayed.

1 12. (Unchanged) The graphics processor of claim 11 further including a load instruction unit
2 for storing load instructions representative of a shortened display list instruction, said load
3 instruction comprising a plurality of data bits each of said plurality of data bits representing
4 specific load functions to be performed by the load instruction.

1 13. (Unchanged) The graphics processor of claim 12, wherein the load instruction unit
2 includes an opcode storage unit for storing opcode information responsive to each of the load

3 instructions for determining the type of function to be performed by the graphics primitive to be
4 rendered.

B²
1 14. (Amended) The graphics processor of claim 13, wherein the load instruction unit further
2 includes a write enable storage portion for storing write enable data for determining whether to
3 load one of the plurality of [registers in the] register files.

1 15. (Unchanged) The graphics processor of claim 14, wherein the load instruction unit
2 further includes an instruction partition portion for storing partition data for referencing the
3 partition table to load parameter values to the referenced register.

B³
1 16. (Amended) The graphics processor of claim [10] 12 further comprising a data shifter
2 coupled to the load instruction unit for sequentially shifting data bits corresponding to a load
3 instruction in order to write the load instructions to the register files.

1 17. (Amended) The graphics processor of claim 16 further comprising an address counter
2 coupled to the register files for sequentially counting [the] address offsets of the register files
3 locations as the load instruction data is loaded into the register files.

1 18. (Unchanged) The graphics processor of claim 17 further comprising a partition table
2 coupled to the load instruction unit for storing the address offset bits corresponding to random
3 register locations in the register files for the display parameter values in the display list.

B⁴
1 19. (Amended) The graphics processor of claim 18 further comprising a write enable signal
2 coupled to [the] an address bit shifter, said write enable signal asserted high to allow the graphics

3 processor to write the load instructions to the register files, wherein the write enable signal
4 enables the graphics processor to randomly load register locations in the register file.

1 20. (Unchanged) The graphics processor of claim 19 further comprising a request next
2 parameter value signal coupled to the fetch logic unit, said request next parameter signal asserted
3 high to allow the next parameter value in the display list to be fetched by the graphics processor.

1 21. (Amended) The graphics processor of claim 20 wherein the partition [look-up] table
2 comprises 64 entries of address offsets to the register file.

1 22. (Unchanged) The graphics processor of claim 21 wherein the 64 entries of the partition
2 table are evenly distributed to corresponding register locations in the register file.

1 23. (Unchanged) The graphics processor of claim 22 wherein each of the 64 entries of the
2 partition table is 6 binary wide.

1 24. (Unchanged) The graphics processor of claim 23 wherein the register file comprises
2 1024 entries of addresses.

1 25. (Canceled) A method of encoding and decoding a shortened display list load instruction
2 comprising the steps of:

3 encoding a field load instruction wherein the field load instruction comprises an opcode
4 instruction, a write enable field and a partition index field;
5 loading the field load instruction to a register file; and
6 executing the field load instruction in a shortened processing time.

1 26. (Canceled) The method of claim 25 wherein the field load instruction execution step
2 comprises the step of enabling the write enable field to allow the load instruction to be randomly
3 load to the register file.

1 27. (Canceled) The method of claim 25 wherein the loading step comprises the step of
2 loading a shift register with write enable data from the field load instruction.

1 28. (Canceled) The method of claim 27 wherein the instruction loading step further
2 comprises the step of indexing a partition look-up table with write enable data from the write
3 enable field.

1 29. (Canceled) The method of claim 27 wherein the instruction loading step further includes
2 loading an address counter to sequentially count the number of load instructions in the display
3 list.

1 30. (Canceled) The method of claim 29 wherein the write enable field is disabled to skip the
2 loading of a register in the register file.